Automated Business Readable Web Tests with Selenium and SpecFlow

Jason Roberts
.NET MVP

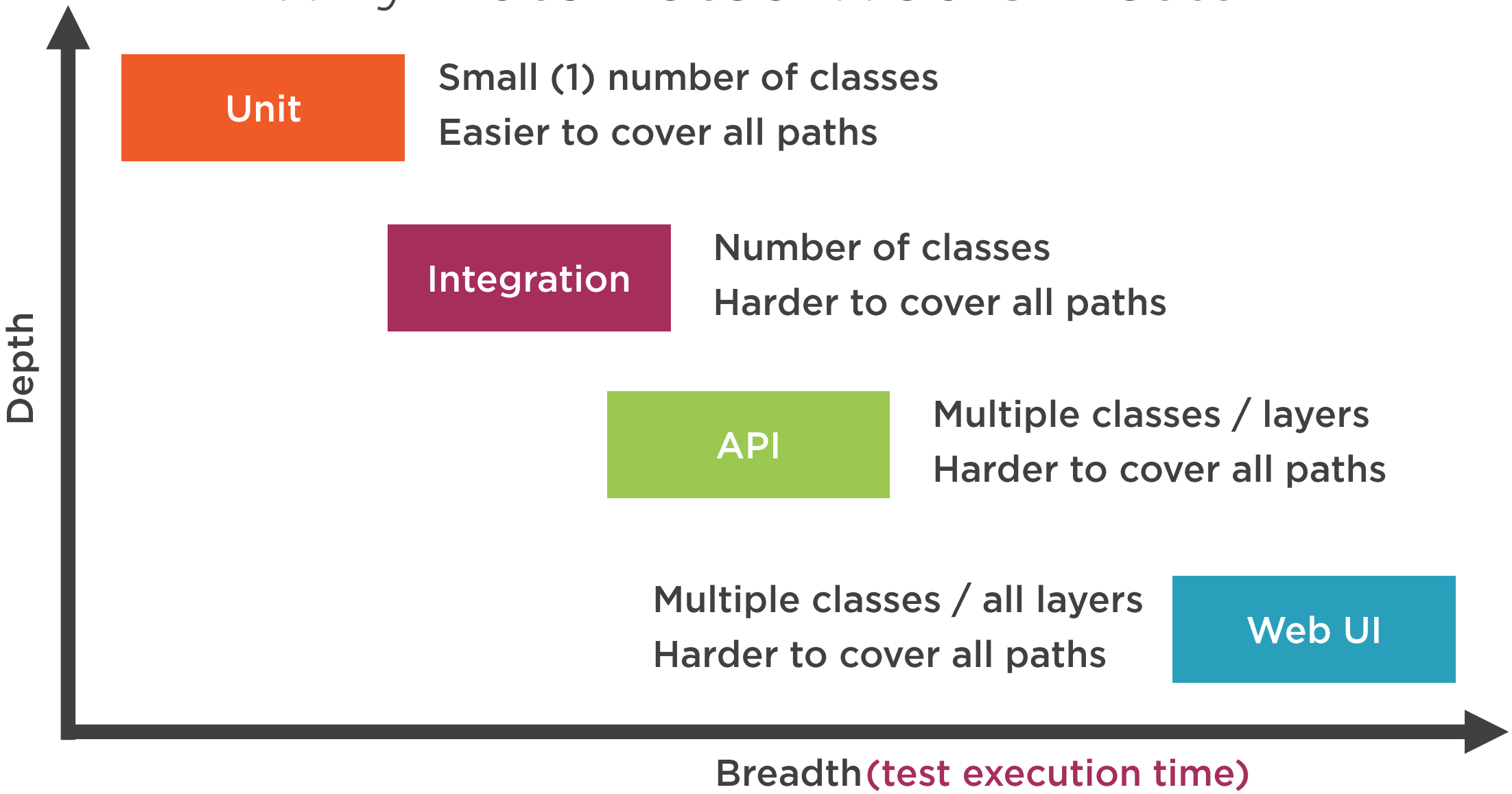
@robertsjason dontcodetired.com

Overview

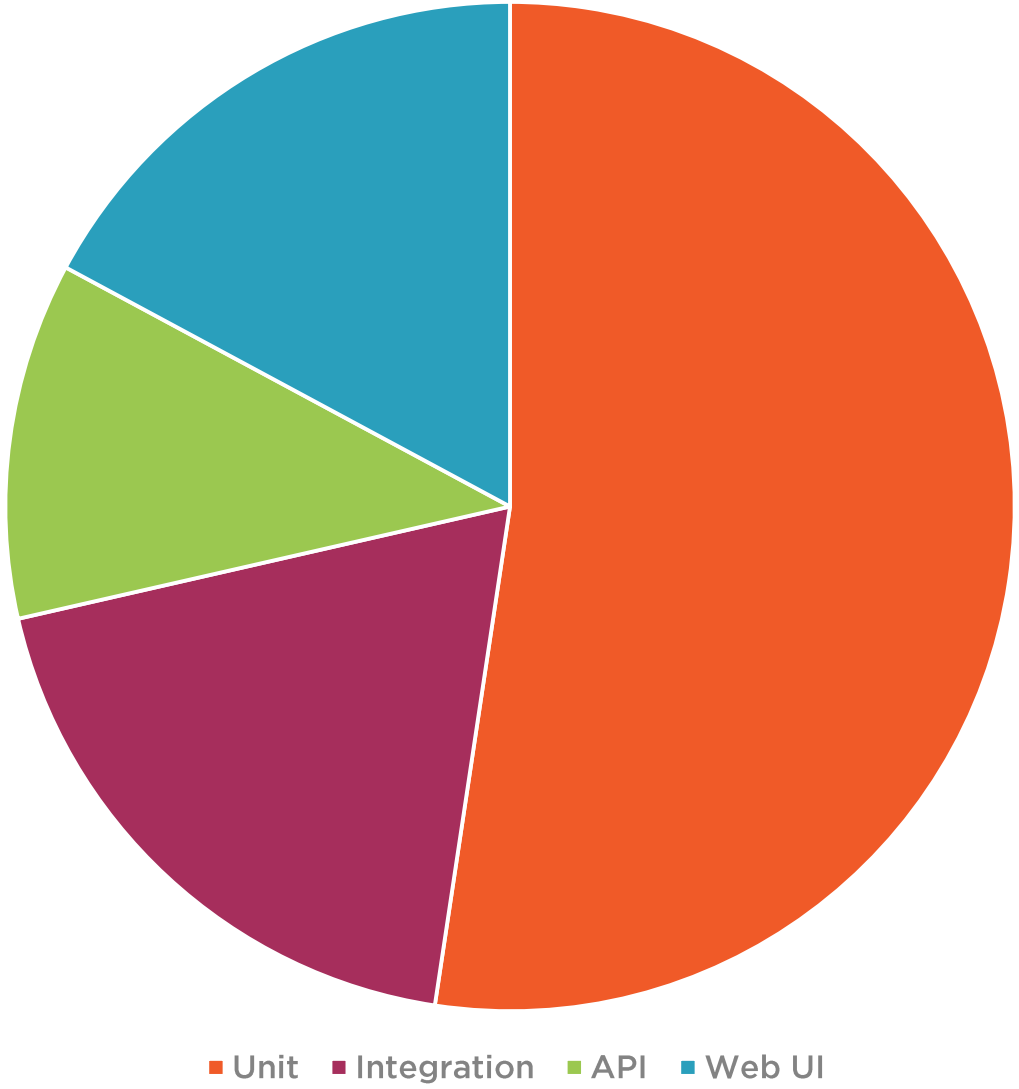
- Why write automated web UI tests?
- Compare to unit, integration, and API tests
- Web UI tests as part of overall test suite
- Automated v. manual UI testing
- Overview of Selenium and SpecFlow
- Bridging the communications gap
- Course demo web site

Course Outline

Why Automated Web UI Tests?



Web Tests as Part of the Overall Test Suite



Why Automated Web UI Tests?

UI databinding errors

Full stack integration errors, e.g.

- Client/server-side validation mismatch
- DB field too small / UI text field
- Serialization, data mapping, conversion

Overall user experience, e.g.

- Application flow
- Validation / help messages shown

Web site configuration errors

(Temporary) substitute for other tests

Run same tests against different browsers

Not usually testing look & feel / design

Automated UI Versus Manual UI Testing

Run faster than human

Quicker feedback

“Free” to run any time

Less error prone

Repetitiveness may introduce human error

Manual tests script need to be maintained

Free up human testers

Better use their skills

Exploratory testing

UI Automation Scenario Selection Considerations

Go where the profit is

Risk mitigation approach

- Legal implications / requirements
- Reputation of organization
- Data corruption
- Security / privacy protection

Most used features

Unique / differentiating features

Any lower level tests?

Total number features / pages

Consider min 1 “smoke test” per page

Introducing Selenium

Selenium

Automate browsers

Simulates a human interacting

Clicking buttons, typing text, etc.

IDE

Record automation test scripts

Playback scripts

Edit scripts

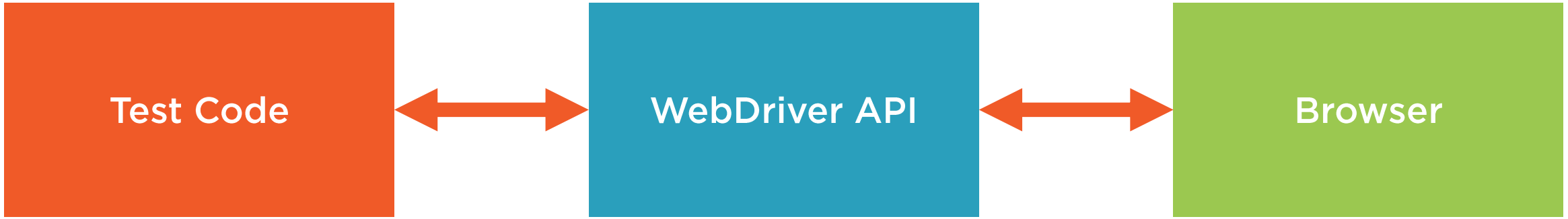
WebDriver

Code-based

More maintainable

Full power of programming languages

Selenium WebDriver



Language bindings

C#, Java, Ruby, etc.

Multiple implementations

Chrome, Firefox, IE, etc.

Selenium WebDriver Examples

Navigate to the home page

Click the button with an ID of “apply”

Type “Sarah” into the name input

Get the text content of the SPAN that has a CSS class of “confirmation”

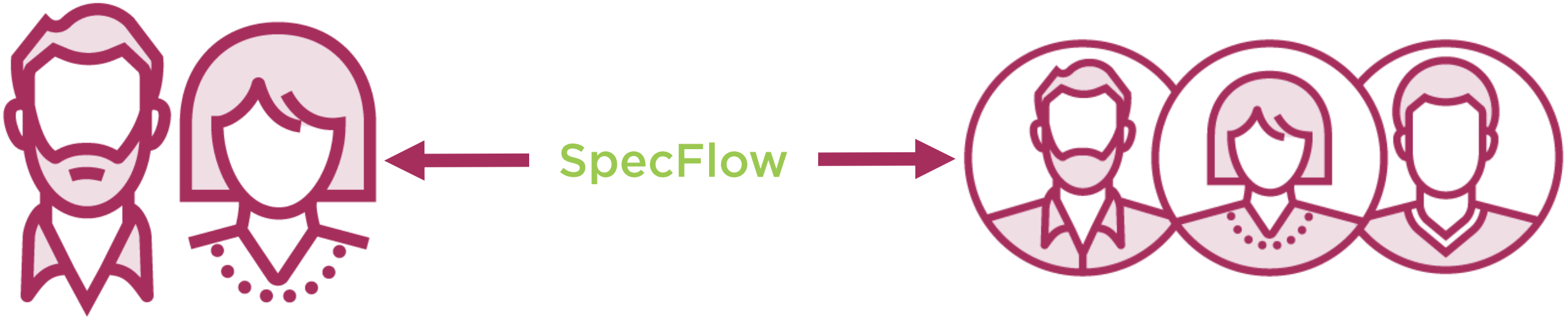
Choose the “Savings” radio button

Check the tick box with an ID of “terms”

Get the title of the current page

Maximize the browser window

Bridging the Communications Gap



Business, client, stakeholders

Developer(s) / development team

SpecFlow

Write tests that the business can easily understand

Better communication between development team and business

Common spoken (non-code) language, e.g. English, French, etc.

Generally non-technical

Ensure correct features are being built

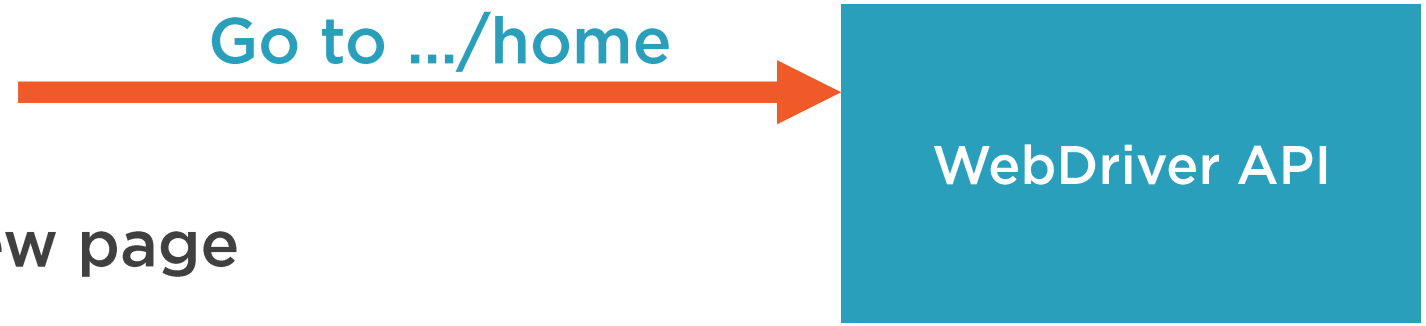
Ensure different scenarios are covered

Use with any type of tests, not just UI

“Business Readable Automated Tests with SpecFlow 2” Pluralsight course

SpecFlow and Selenium

Given I'm on the home page
When I choose more info
Then I should be taken to new page



SpecFlow and Selenium

SpecFlow and Selenium

Given I'm on the home page
When I choose more info
Then I should be taken to new page

Summary

- Why write automated web UI tests?
- Full stack integration tests
- Compliment unit, integration, and API tests
- Automated v. manual UI testing
- Quicker and free up human testers
- Selenium WebDriver to automate the browser from test code
- SpecFlow to bridge the communications gap
- Course demo web site

Next:

Getting Started with Selenium